Construction of Parametrically-Robust CFD-Based Reduced-Order Models for PDE-Constrained Optimization

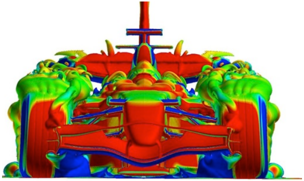
Matthew J. Zahr, David Amsallem, Charbel Farhat

Farhat Research Group
Stanford University

25th June 2013
San Diego, CA
43rd AIAA Fluid Dynamics Conference and Exhibit

Motivation

- Complex, steady-state problems



- Real-time analyses
  - Model Predictive Control
- Many-query analyses
  - Optimization
  - Uncertainty Quantification
  - Routine Analysis

Projection-based Model Reduction

- High-Dimensional Model (HDM)
  - Large-scale discretization of Navier-Stokes equation
  - Parametric: shape parameter, Mach number, angle of attack
- Reduced-Order Model (ROM)
  - Offline
    - Sample HDM at multiple parameter configurations (training set)
    - Collect snapshots
    - Compress snapshots → Reduced-Order Basis (ROB) → reduce dimension of state vector
    - Project governing equations onto a related subspace
  - Online
    - Query ROM

ROM Optimization Applications

- For the use of ROMs as surrogates in optimization, the "offline" cost must not be too large
  - Only timing important is the clock-time from the beginning of the design process to the time the solution is obtained
  - No distinction between offline cost and online cost
- Well-known that ROMs lack robustness away from their training configurations
- Sample HDM only in regions of interest
  - Vicinity of current iterate
  - Vicinity of optimal solution
- Such regions not known apriori, i.e. in the "offline" phase
- Offline/online framework not natural in optimization context

High-Dimensional Model (HDM)

We are interested in solving the following discretized, steady-state PDE:

$$\mathbf{R}(\mathbf{w}; \boldsymbol{\mu}) = 0,$$

where $\mathbf{w} \in \mathbb{R}^N$ is the state vector (N typically very large) and $\boldsymbol{\mu} \in \mathbb{R}^p$ is the vector of parameters.

Examples

- Navier-Stokes equations
- Euler equations

Reduced-Order Model (ROM)

- Define a ROB $\boldsymbol{\Phi}^{\mathcal{D}} \in \mathbb{R}^{N \times n_y}$ generated from snapshots of the HDM in the training set $\mathcal{D} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_{n_s}\}$
- Model Order Reduction (MOR) assumption

$$\mathbf{w} = \bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}} \mathbf{w}_r$$

where $\mathbf{w}_r \in \mathbb{R}^{n_y}$ are the reduced coordinates and $n_y \ll N$.

- N equations, n_Y unknowns

$$\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}} \mathbf{w}_r; \boldsymbol{\mu}) \approx 0$$

- Require residual $\perp$ to the subspace spanned by a left basis, $\boldsymbol{\Psi} \in \mathbb{R}^{N \times n_Y}$

$$\boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}} \mathbf{w}_r; \boldsymbol{\mu}) = 0$$

Reduced-Order Model (ROM)

- Define a ROB $\mathbf{\Phi}^{\mathcal{D}} \in \mathbb{R}^{N \times n_y}$ generated from snapshots of the HDM in the training set $\mathcal{D} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_{n_s}\}$
- Model Order Reduction (MOR) assumption

$$\mathbf{w} = \bar{\mathbf{w}} + \mathbf{\Phi}^{\mathcal{D}} \mathbf{w}_r$$

  where $\mathbf{w}_r \in \mathbb{R}^{n_y}$ are the reduced coordinates and $n_y \ll N$.
- N equations, n_Y unknowns

$$\mathbf{R}(\bar{\mathbf{w}} + \mathbf{\Phi}^{\mathcal{D}} \mathbf{w}_r; \boldsymbol{\mu}) \approx 0$$

- Require residual $\perp$ to the subspace spanned by a left basis, $\mathbf{\Psi} \in \mathbb{R}^{N \times n_Y}$

$$\boxed{\mathbf{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \mathbf{\Phi}^{\mathcal{D}} \mathbf{w}_r; \boldsymbol{\mu}) = 0}$$

Hyperreduction

- Despite reduced dimension, still requires operations scaling with N

$$\boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}} \mathbf{w}_r; \boldsymbol{\mu}) = 0$$

- Build reduced basis for residual $\boldsymbol{\Phi}_r \in \mathbb{R}^{N \times n_r}$

$$\mathbf{R} \approx \boldsymbol{\Phi}_r \mathbf{R}_r$$

- Determine residual reduced coordinates via *gappy* approximation

$$\mathbf{R}_r = \arg \min_{\mathbf{r} \in \mathbb{R}^{n_r}} \left|\left| \mathbf{Z}^T \mathbf{R} - \mathbf{Z}^T \boldsymbol{\Phi}_r \mathbf{r} \right|\right|$$

where $\mathbf{Z} \in \mathbb{R}^{N \times n_i}$ is a restriction operator

- Similar procedure for Jacobian
- Avoids *online* operations scaling with N

ROM-Constrained Optimization

PDE-Constrained Optimization

$$\begin{aligned} &\underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} && f(\mathbf{w}(\boldsymbol{\mu}),\boldsymbol{\mu}) \\ &\text{subject to} && \mathbf{c}(\mathbf{w}(\boldsymbol{\mu}),\boldsymbol{\mu}) \leq 0 \end{aligned} \tag{1}$$

$\mathbf{w}$ is implicitly defined as a function of $\boldsymbol{\mu}$ via the equation $\mathbf{R}(\mathbf{w};\boldsymbol{\mu}) = 0$.

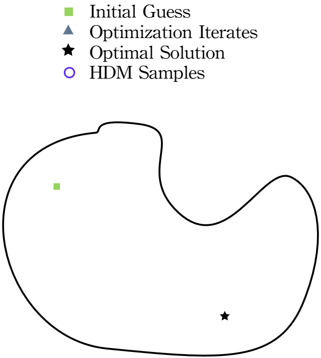
ROM-Constrained Optimization

$$\begin{aligned} &\underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} && f(\bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}}\mathbf{w}_r(\boldsymbol{\mu}),\boldsymbol{\mu}) \\ &\text{subject to} && \mathbf{c}(\bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}}\mathbf{w}_r(\boldsymbol{\mu}),\boldsymbol{\mu}) \leq 0 \end{aligned} \tag{2}$$

$\mathbf{w}_r$ is implicitly defined as a function of $\boldsymbol{\mu}$ via the equation $\boldsymbol{\Psi}^T\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}}\mathbf{w}_r;\boldsymbol{\mu}) = 0$.

Progressively Updated ROMs for Optimization

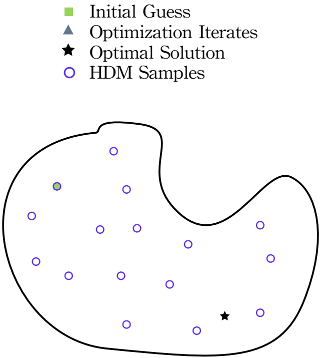
Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



Sample HDM Apriori

Progressively Updated ROMs for Optimization

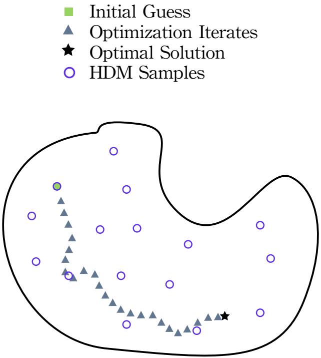
Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



Sample HDM Apriori

Progressively Updated ROMs for Optimization

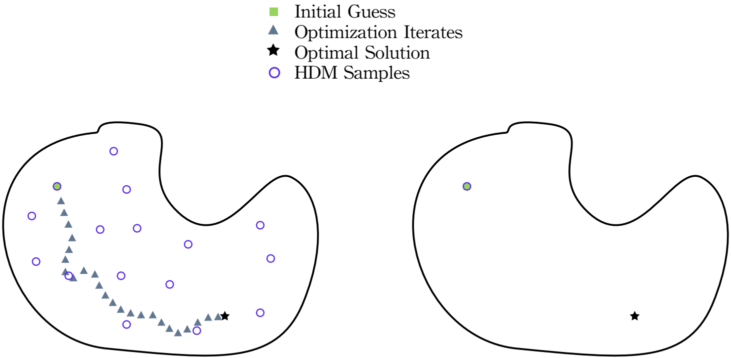
Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



Sample HDM Apriori

Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest

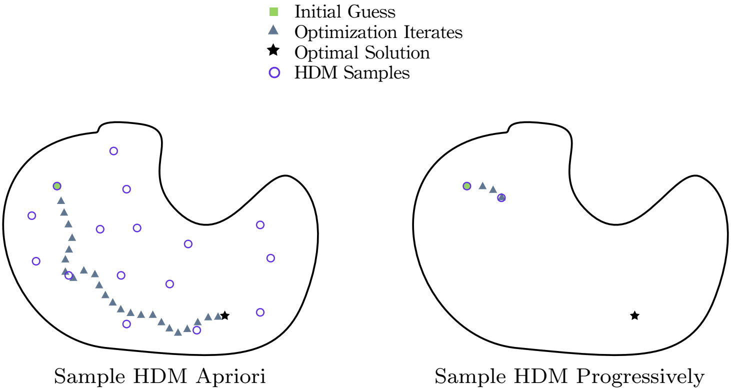


Sample HDM Apriori

Sample HDM Progressively

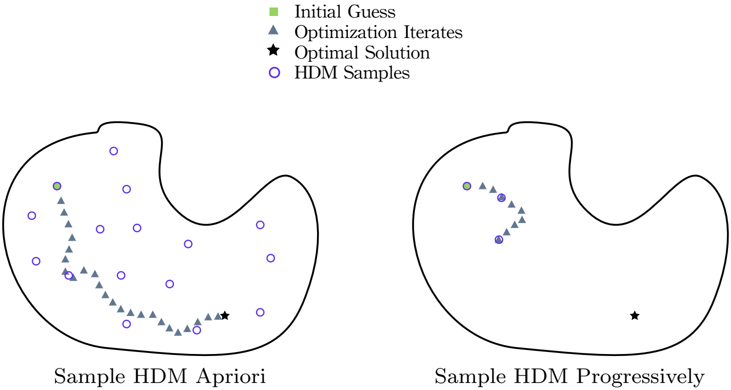
Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



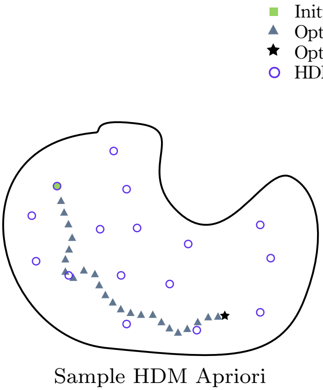
Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



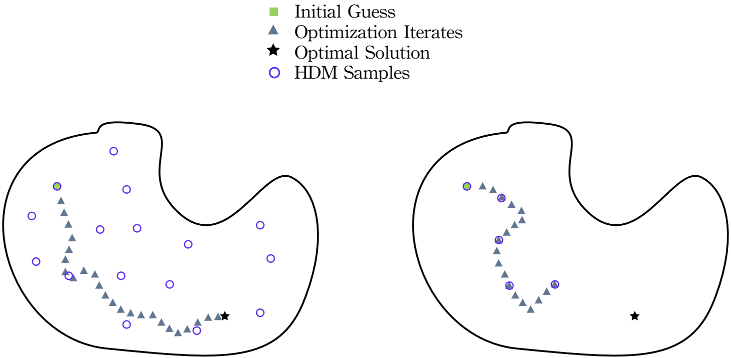
Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest

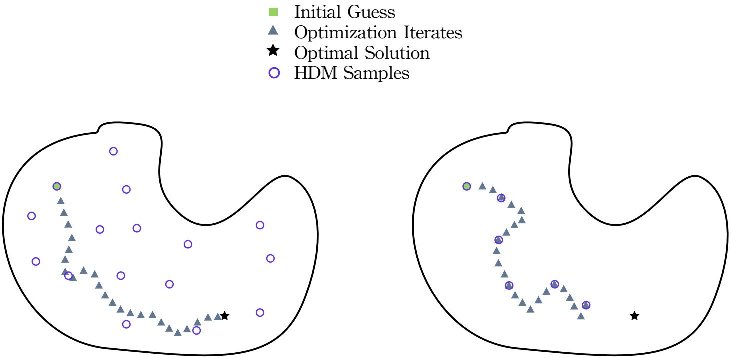


Sample HDM Apriori

Sample HDM Progressively

Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest

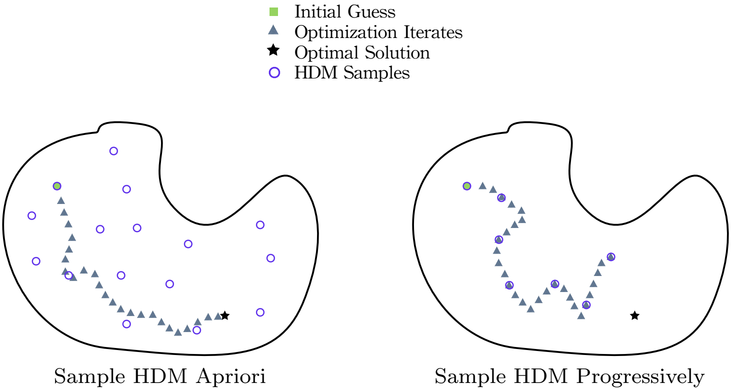


Sample HDM Apriori

Sample HDM Progressively

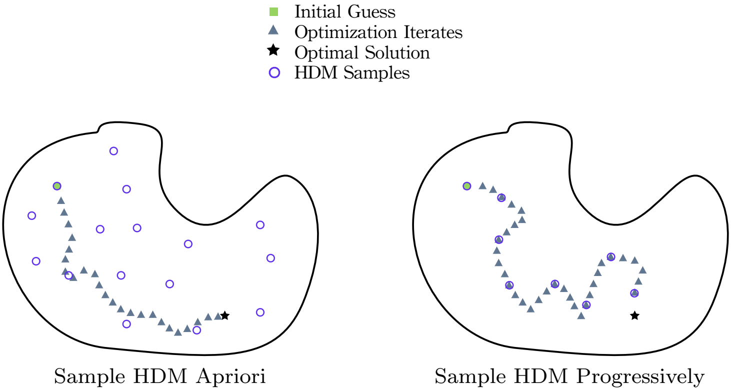
Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



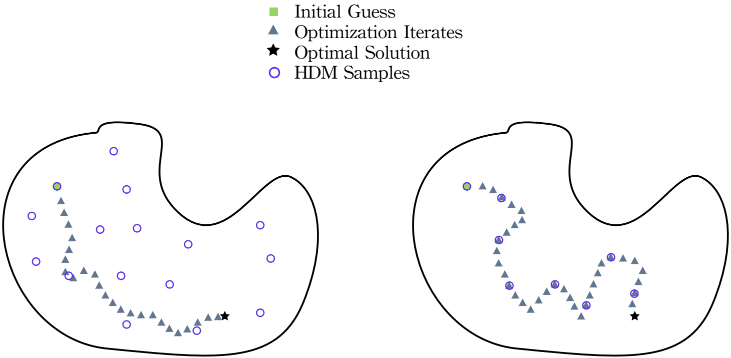
Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



Progressively Updated ROMs for Optimization

Progressively construct a ROM specialized to a particular region of the parameter space while using it to solve the optimization problem of interest



Sample HDM Apriori

Sample HDM Progressively

Sequential Optimization Subproblem

Given $\mathbf{\Phi}^{\mathcal{D}}$ generated from a very coarse sampling of parameters $\mathcal{D} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k\}$, solve

$$\begin{aligned} \underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f(\bar{\mathbf{w}} + \mathbf{\Phi}^{\mathcal{D}}\mathbf{w}_r(\boldsymbol{\mu})) - \frac{\gamma}{2}\left|\left|\mathbf{R}(\bar{\mathbf{w}} + \mathbf{\Phi}^{\mathcal{D}}\mathbf{w}_r(\boldsymbol{\mu}); \boldsymbol{\mu})\right|\right|_2^2 \\ \text{subject to} \quad & \mathbf{c}(\bar{\mathbf{w}} + \mathbf{\Phi}^{\mathcal{D}}\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) \leq 0. \end{aligned} \tag{3}$$

$\mathbf{w}_r(\boldsymbol{\mu})$ is implicitly defined via $\mathbf{\Psi}^T\mathbf{R}(\bar{\mathbf{w}} + \mathbf{\Phi}^{\mathcal{D}}\mathbf{w}_r; \boldsymbol{\mu}) = 0$.

- *Cost*: Every iteration requires ROM solution.
- Does not require solution of HDM until sample is found

ROM-Constrained Sampling Algorithm

Algorithm 1 Progressively Updated ROMs for Optimization

Input: Initial ROB, $\boldsymbol{\Phi}^{\mathcal{D}}$; $\gamma_0 \in \mathbb{R}_+$; $0 < \rho < 1$
Output: Solution to (2), $\boldsymbol{\mu}^*$ (approximation to solution of (1))

$\gamma = \gamma_0$
while optimality conditions of (2) not satisfied **do**
  Solve (3) $\rightarrow \boldsymbol{\mu}^*$ and $\mathbf{w}_r^*$
  if $||\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{w}_r^*, \boldsymbol{\mu}^*)|| > \epsilon$ **then**
    Solve $\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}^*) = 0 \rightarrow \mathbf{w}^*$
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{\boldsymbol{\mu}^*\}$
    Update $\boldsymbol{\Phi}^{\mathcal{D}}$ with new snapshots $\mathbf{w}^*$
  end if
  $\gamma = \rho\gamma$
end while

Quasi-1D Euler Flow

Quasi-1D Euler equations:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{1}{A}\frac{\partial (A\mathbf{F})}{\partial x} = \mathbf{Q}$$

where

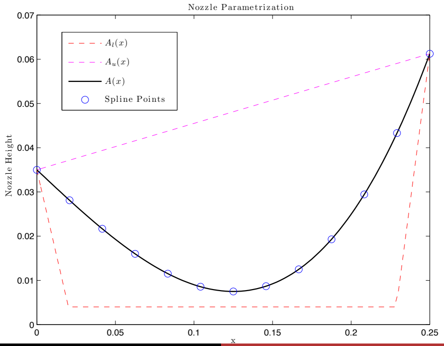
$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \qquad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ (e+p)u \end{bmatrix}, \qquad \mathbf{Q} = \begin{bmatrix} 0 \\ \frac{p}{A}\frac{\partial A}{\partial x} \\ 0 \end{bmatrix}$$

- Semi-discretization $\Longrightarrow$ finite volumes with Roe flux and entropy corrections
- Full discretization $\Longrightarrow$ Backward Euler $\rightarrow$ steady state

Nozzle Parametrization

Nozzle parametrized with *cubic splines* using 13 control points and constraints requiring

- convexity $A''(x) \geq 0$
- bounds on $A(x)$ $A_l(x) \leq A(x) \leq A_u(x)$
- bounds on $A'(x)$ at inlet/outlet $A'(x_l) \leq 0,\ A'(x_r) \geq 0$



Numerical Experiment (Parameter Estimation)

Select $\boldsymbol{\mu}^{ex}$ and let $\mathbf{w}^{ex}$ be the solution of $\mathbf{R}(\mathbf{w};\boldsymbol{\mu}^{exact})$, then the parameter estimation problem is

$$\begin{aligned} \underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f(\boldsymbol{\mu}) = \frac{1}{2}\left|\left|\bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}}\mathbf{w}_r(\boldsymbol{\mu}) - \mathbf{w}^{ex}\right|\right|_2^2 \\ \text{subject to} \quad & \mathbf{c}(\bar{\mathbf{w}} + \boldsymbol{\Phi}^{\mathcal{D}}\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) \leq 0 \end{aligned} \tag{4}$$

Optimization Comparison

- **HDMOpt**, whereby (4) is solved with standard PDE-constrained optimization technology (NAND framework) without introducing a reduced-order model,
- **LHSOpt**, whereby a ROM is built using standard techniques with parameter samples chosen based on a Latin Hypercube Sampling (LHS) of the feasible region; the HDM is replaced with the resulting ROM in (4),
- **MergeOpt**, whereby Algorithm 1 is applied to (4).

ROM Optimization Experiment Results

Table : Accuracy and performance comparison for various PDE-constrained optimization methods for the shape optimization problem

	MergeOpt	**LHSOpt** [1]	**HDMOpt**
$\boldsymbol{\mu}^{ex}$ **Error (%)**	2.88	(3.32, 44.7, 16.4)	1.04×10^{-7}
$\mathbf{w}^{ex}$ **Error (%)**	0.67	(0.84, 89.6, 12.3)	6.79×10^{-9}
# HDM	9	(12,12,12)	202
# ROM	770	(2, 613, 168)	-

[1] (min, max, mean)

Experiment Results: LHSOpt

Figure : LHSOpt Nozzle Configuration Samples

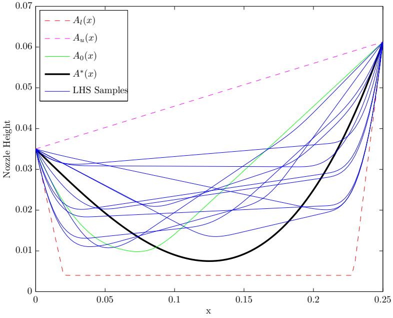
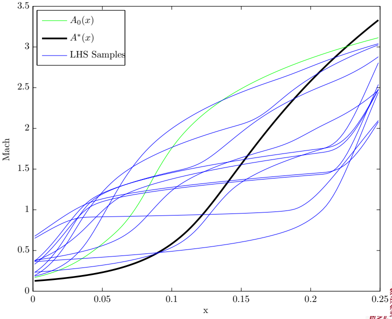


Figure : Mach Distribution at LHSOpt Samples



Experiment Results: MergeOpt

Figure : MergeOpt Nozzle Configuration Samples

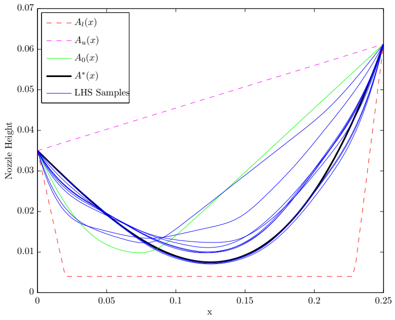
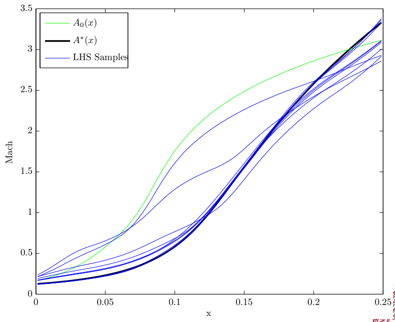


Figure : Mach Distribution at MergeOpt Samples



Conclusions and Future Work

- Proposed approach progressively updates a ROM while solving a PDE-constrained optimization problem using information regarding:
  - the physics of the problem
  - current state of the ROM
- Proposed approach *specializes* the ROM to the vicinity of the current iterate instead of attempting to make the ROM accurate in the entire parameter space
  - Requires fewer HDM queries than HDMOpt and LHSOpt
  - Achieves low errors in $\boldsymbol{\mu}^{ex}$ and $\mathbf{w}^{ex}$
- Hyperreduction necessary to realize speedups

Acknowledgments

Department of Energy Computational Science Graduate Fellowship